基于深度序列加权核极限学习的入侵检测算法 *

汪 洋，伍忠东，朱 婧

(兰州交通大学，电子与信息工程学院，兰州 730070)

摘 要：针对海量多源异构且数据分布不平衡的网络入侵检测问题以及传统深度学习算法无法根据实时入侵情况在线更新其输出权重的问题，提出了一种基于深度序列加权核极限学习的入侵检测算法(DBN-WOS-KELM 算法)。该算法先使用深度信念网络 DBN 对历史数据进行学习，完成对原始数据的特征提取和数据降维，再利用加权序列核极限学习机进行监督学习完成入侵识别，结合了深度信念网络提取抽象特征的能力以及核极限学习机的快速学习能力。最后在 KDD99 部分数据集上进行了仿真实验，实验结果表明 DBN-WOS-KELM 算法提高了对小样本攻击的识别率，并且能够根据实际情况在线更新输出权重，训练效率更高。

关键词：深度信念网络；序列学习；核极限学习；样本加权；入侵检测

中图分类号：TP393.08 doi: 10.19734/j.issn.1001-3695.2018.08.0653

Intrusion detection algorithm based on depth sequence weighted kernel extreme learning

Wang Yang, Wu Zhongdong, Zhu Jing

(*School of Electronic & Information Engineering, Lanzhou Jiaotong University, Lanzhou* 730070, *China*)

Abstract: This paper proposed a intrusion detection algorithm based on deep sequence weighting kernel limit learning (DBN-WOS-KELM) to solve the problem of massive multi-source heterogeneous network intrusion detection with unbalanced data distribution and the problem that the traditional deep learning algorithm can not update its output weight online according to the real-time intrusion situation. The algorithm first uses the deep belief network DBN to study the historical data, then extracts the features of the original data and reduces the dimension of the data, and then uses the weighted sequence kernel extreme learning machine for supervised learning to complete the intrusion detection. It combines the ability of extracting Abstract features from the deep belief network and the fast learning ability of the kernel extreme learning machine. Finally, the simulation experiments on KDD99 dataset show that DBN-WOS-KELM algorithm improves the recognition rate of small sample attacks, and can update the output weights online according to the real-time situation, so that the training efficiency is much higher.

Key words: deep belief network; sequence learning; kernel extreme learning; sample weighting; intrusion detection

0 引言

入侵检测技术是信息网络领域不可缺少的部分，随着人工智能技术的不断深入，深度学习算法相对于传统的机器学习入侵检测算法具有识别率高，误报率低的优势，在入侵检测领域应用广泛。高妮等人[1]提出了深度信念网络的入侵检测算法,Ambusaidi 等人[2]提出了基于 DBN-SVM 的混合入侵检测算法，逯玉婧、杨昆朋[3,4]提出了基于深度学习的混合入侵检测模型，得到的实验结果在准确率和误报率上均优于传统的机器学习入侵检测算法。但是现阶段学习算法未考虑到参与训练的网络历史入侵数据分布的不平衡性，只强调了高检测率和低误报率，导致了小样本攻击类别大部分被识别为大样本攻击类别，对小样本攻击类别的检测准确率不高。针对此类不平衡数据的学习分类问题，Zong 等人[5]提出了加权极限学习算法处理不平衡数据。Mirza 在研究了序列学习的优势，提出了加权在线极限学习机的不平衡数据学习算法以及加权的序列核极限学习算法[6,7],实验结果表明小样本数据识别大幅度增加。本文针对网络入侵数据的多源异构以及各攻击类别的分布不平衡特性以及传统深度学习算法无法根据实时入侵数据在线更新其输出权重的问题，对深度信念网络和序列核极限学习机进行了深入研究，提出了基于 DBN-WOS-KELM 的入侵检测算法，该算法充分利用了 DBN 提取数据特征的能力和序列核极限学习机的泛化能力。最后通过 KDD99 部分数据集进行有效评估，实验结果表明，本文算法不仅提高了对小样本攻击的检测率，还能根据实际情况在线更新分类器参数，进一步提高了训练效率。

1 DBN-WOS-KELM 入侵检测模型

1.1 方法总体架构

本文提出了 DBN-WOS-KELM 的入侵检测算法，总体框架如图 1 所示，具体步骤如下：

a)数据预处理。将网络数据中的字符特征转换成对应的二进制数据，再均值化到[0,1]。

b)DBN 抽象特征提取。分为 RBM 的无监督预训练以及 BP 微调全局，对网络数据进行降维。

c)WOS-KELM 分类器识别入侵。将降维后的数据打上数据标签作为可靠数据并进行样本加权，通过 KELM 的序列学习，学习完成后取代 BP 作为分类器。

收稿日期：2018-08-26；**修回日期：**2018-10-12 **基金项目：**甘肃省高等学校创新团队项目（2017C-09）；中国铁路总公司科技研究开发计划重大课题（2017X013-A）

作者简介：汪洋（1994-），男，四川资阳人，硕士研究生，主要研究方向为深度学习、信息安全；伍忠东（1968-），男，湖南醴陵人，教授，硕导，主要研究方向为信息网络安全算法（WUZHD@mail.lzjtu.cn）；朱婧（1993-），女，河南漯河人，主要研究方向为深度学习、SDN.

d)DBN-WOS-KELM 识别入侵。将网络历史入侵数据降维后进行样本加权后进行入侵检测，识别出其入侵类型。

图 1　DBN-WOS-KELM 总体框架

Fig. 1　The overall framework of DBN-WOS-KELM

1.2　DBN 特征降维

深度信念网络是一种由多层 RBM 和一层 BP 网络构成的生成性深度结构，最先由 Hinton 提出[8]，如图 1 步骤 b）所示，其训练过程可分解如下两步：

1) RBM 预训练

RBM 的结构如图 2 所示，分为输入层和隐藏层两层。其参数为 $\theta=\{W,b,c\}$ 。

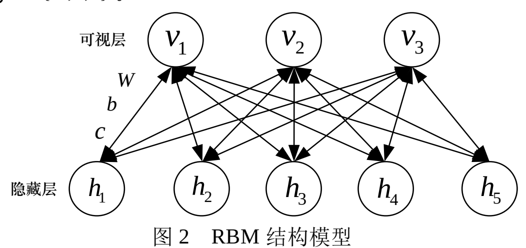


图 2　RBM 结构模型

Fig. 2　The structural model of RBM

其中 b 是输入层到隐藏层的偏置， c 是隐藏层到输入层的偏置， W 是可见层到隐藏层节点的连接权重矩阵， v_i 和 h_j 是分别表示两层神经元的状态。RBM 的训练算法是 Hinton 提出的对比散度学习算法[9][10]，算法过程如下：

算法 1　对比散度学习算法

输入：训练样本 x_0 ；隐藏层神经元个数 m ；学习率 ε ；最大迭代次数 T 。

输出：RBM 参数 $\theta=\{W,b,c\}$ 。

训练阶段： $v_0=x_0$ ；随机初始化 θ

对所有隐藏层单元：

计算 $p(h_{1j}=1|v_1)=\sigma(b_j+\sum_i v_{1i}W_{ij})$

从条件分布 $p(h_{1j}=1|v_1)$ 中抽取 $h_{1j}\in(0,1)$

对所有可见层单元：

计算 $p(v_{2i}=1|h_1)=\sigma(c_i+\sum_i W_{ij}h_{1j})$

从条件分布 $p(v_{2i}=1|h_1)$ 中抽取 $v_{2i}\in(0,1)$

对所有隐藏层单元：

计算 $p(h_{2j}=1|v_2)=\sigma(b_j+\sum_i v_{2i}W_{ij})$

更新参数：

$$W\leftarrow W+\varepsilon(p(h_1=1|v_1)v_1^T-p(h_2=1|v_2)v_2^T)$$

$$c\leftarrow c+\varepsilon(v_1-v_2)$$

$$b\leftarrow c+\varepsilon(p(h_1=1|v_1)-p(h_2=1|v_2))$$

2) 权值微调

为使整个 DBN 网络达到全局最优，通过 BP 网络利用少量真实可靠的标签数据微调各层 RBM 参数，将预测值与实际标签对比得到误差，并将误差传播至每一层，具体步骤如算法 2 所示。

算法 2　反向传播算法

输入：预训练后的 DBN 参数 $\theta=\{W,b,c\}$ ，训练样本 $<v_i, t_i>$，最大迭代次数 T 。

输出：微调后的 DBN 参数 $\theta'=\{W,b,c\}$ 。

训练阶段：对每一个样本 v_i 计算 DBN 的重构输出 v_i' ，反向传播误差；

对每个输出单元计算误差 δ_k ：

$$\delta_k=v_k'(1-v_k')(v_k-v_k')$$

对每个隐藏层单元计算误差 δ_h ：

$$\delta_h=v_h'(1-v_h')\sum_{k\in outputs}\theta_{kh}\delta_k$$

更新参数：

$$\theta_{ji}=\theta_{ji}+\Delta\theta_{ji}$$

其中 $\Delta\theta_{ji}=\eta\delta_j x_j$ ， η 为学习率。

1.3　WOS-KELM 分类器

加权序列和极限学习机(WOS-KELM)[7]是加权序列学习算法与加权核极限学习机算法(WKELM)[5]的结合，由 Shuya Ding 等人提出。与序列核极限学习机(OS-KELM)[11]类似，训练过程分成初始化阶段和序列学习阶段两个阶段。

1)初始化阶段

WOS-KELM 的初始化阶段与 WELM 学习算法类似，首先选择初始训练数据 $n_0=(x_i,t_i)$ $i=1,\dots,N_0$ 。确定隐藏层初始化输出矩阵为

$$\beta^{(0)}=(H_0H_0^T+\frac{W_0^{-1}}{C})^{-1}T_0=R_0T_0 \tag{1}$$

$$R_0=(H_0H_0^T+\frac{W_0^{-1}}{C})^{-1} \tag{2}$$

$$W_1^{-1}=diag\begin{pmatrix}\frac{1}{w_1^1} & \frac{1}{w_2^1} & \cdots & \frac{1}{w_{N_0}^1}\end{pmatrix} \tag{3}$$

定义 $H_0H_0^T=\Omega_{ELM}$ ， $\Omega_{ELMij}=K(x_i,x_j)$ ， $K(x,y)$ 为核函数， T_0 为样本标签矩阵， C 为正则化系数， W_0 为第一批训练样本的权重矩阵，文献[7]给出了 W 的计算方法

$$W_k=diag\begin{pmatrix}w_{k,1} & w_{k,2} & \cdots & w_{k,N_k}\end{pmatrix} \tag{4}$$

$$w_{k,i}=1/m_i \tag{5}$$

式(4)中 N_k 表示训练样本的数量， $x_{k,i}$ 表示第 k 条属于第 i 类标签的训练样本， m_i 表示训练数据中第 i 类标签的样本数目。

2)序列学习阶段

当第一批训练数据 $n_1=(x_i^1,t_i^1)$ ， $i=1,2,\dots,N_1$ 参与训练时，更新核极限学习机的输出矩阵 $\beta^{(0)}$ 为

$$\beta^{(1)}=\begin{bmatrix}H_0H_0^T+\frac{W_0^{-1}}{C} & H_0H_1^T\\ H_1H_0^T & H_1H_1^T+\frac{W_1^{-1}}{C}\end{bmatrix}^{-1}\begin{bmatrix}T_0\\T_1\end{bmatrix} \tag{6}$$

$$W_1^{-1}=diag\begin{pmatrix}\frac{1}{w_1^1} & \frac{1}{w_2^1} & \cdots & \frac{1}{w_{N_0}^1}\end{pmatrix} \tag{7}$$

定义 $K_0=H_0H_0^T$ ， $K_0'=H_0H_1^T$ ， $K_0''=H_1H_1^T$,则

$$R_1=\begin{bmatrix} K_0+\frac{W_0^{-1}}{C} & K'_0 \\ (K'_0)^T & K''_0+\frac{W_1^{-1}}{C}\end{bmatrix} =\begin{bmatrix} R_0 & O_{N_0,N_1} \\ O_{N_1,N_0} & O_{N_1,N_1}\end{bmatrix}+\begin{bmatrix} -R_0K'_0 \\ I_{N_1}\end{bmatrix} \times(K''_0+\frac{W_1^{-1}}{C}-(K'_0)^T R_0 K'_0)[(-R_0K'_0)^T \quad I_{N_1}] \tag{8}$$

式(8)中 $O_{m,l}$ 表示 $m\times l$ 阶零矩阵，I_m 代表 $m\times m$ 阶单位矩阵。

当第 $k+1$ 批训练数据参与训练时，

$$R_{k+1}=\begin{bmatrix} R_k & O_{N_k,N_{k+1}} \\ O_{N_{k+1},N_k} & O_{N_{k+1},N_{k+1}}\end{bmatrix}+\begin{bmatrix} -R_kK'_k \\ I_{N_{k+1}}\end{bmatrix}\times (K''_k+\frac{W_{k+1}^{-1}}{C}-(K'_k)^T R_k K'_k)^{-1}[(-R_kK'_k)^T \quad I_{N_{k+1}}] \tag{9}$$

$$R_k=(K_k+\frac{W_k^{-1}}{C})^{-1} \tag{10}$$

为了表达简便，定义

$$Q_{k+1}=-R_kK'_k \tag{11}$$

$$\mu_{k+1}=(K''_k+\frac{W_{k+1}^{-1}}{C})+(K'_k)^T Q_{k+1} \tag{12}$$

将式(11)(12)代入式(9)中得到

$$R_{k+1}=\begin{bmatrix} R_k & O_{N_k,N_{k+1}} \\ O_{N_{k+1},N_k} & O_{N_{k+1},N_{k+1}}\end{bmatrix}+\begin{bmatrix} Q_{k+1} \\ I_{N_{k+1}}\end{bmatrix} \times(\mu_{k+1})^{-1}[Q_{k+1}^T \quad I_{N_{k+1}}] \tag{13}$$

最后得到第 $k+1$ 批次输出矩阵的更新

$$\beta^{(k+1)}=R_{k+1}\begin{bmatrix} \breve{T}_k \\ T_{k+1}\end{bmatrix} \tag{14}$$

$$\breve{T}_k=[T_0 \quad \cdots \quad T_k]^T \tag{15}$$

将式(14)(15)代入式(13)中得到

$$\beta^{(k+1)}=\begin{bmatrix} \beta^{(k)}+Q_{k+1}(\mu_{k+1})^{-1}(T_{k+1}-(K'_k)^T\beta^{(k)}) \\ (\mu_{k+1})^{-1}(T_{k+1}-(K'_k)^T\beta^{(k)})\end{bmatrix} \tag{16}$$

2　实验

2.1　数据集预处理

本次实验采用的数据集是 KDD99 数据集[12]，数据集将异常类型分成了四大类，分别是 DOS、R2L、U2R、Probe，含有 39 种攻击方式。训练集中包含了 22 种攻击方式，测试集中还包含了 17 种训练集中未出现的攻击方式（实验未使用）。

数据集有 41 维特征，其中包含了字符型和数字型，在训练和检测前先进行预处理，首先将字符数据转换成二进制向量。例如标签类型 normal、DOS、R2L、U2R、Probe 表示为 [0,0,0,0,1]、[0,0,0,1,0]、[0,0,1,0,0]、[0,1,0,0,0]、[1,0,0,0,0]。字符数据映射完成之后进行数据均值化处理，将各维数据归一化到 0 到 1 之间，转换公式如下：

$$y=\frac{y-MIN}{MAX-MIN} \tag{17}$$

2.2　实验评价标准

传统的入侵检测算法通常强调整体高准确率，以及低误报率，忽略了小样本攻击的检测，本文实验除采用准确率 AC 以及误报率 FA 作为各类攻击检测评价标准以外，还采用几何平均数 G_{mean} [13]作为整体检测结果的评价标准，如式(18)~(20)所示。

$$AC=\frac{T_p+T_N}{T_p+T_N+F_p+F_N} \tag{18}$$

$$FA=\frac{F_p}{T_N+F_p} \tag{19}$$

$$G_{mean}=(\prod_{j=1}^{q}\frac{s_{jj}}{s_j})^{\frac{1}{q}} \tag{20}$$

其中：T_N 表示正常样本正确分类的个数，T_P 表示攻击样本正确分类的个数，F_p 表示正常样本误报为攻击的个数，F_N 表示攻击样本误报为正常的个数，s_j 表示第 j 类攻击样本总数，s_{jj} 表示检测准确的 j 类攻击数，q 表示样本类别数,实验数据共 5 类，故 q 取值为 5。

2.3　实验参数设置

本文所提出的模型中，DBN 作为数据降维的部分，其网络层数的设定至关重要，因此实验前必须确定其深度。实验中预设了 5 种 DBN^i 结构(i 表示网络的层数)，分类器使用 BP 网络，其节点参数如表 1 所示，从 KDD99 的训练集中抽取 20%，测试集中抽取 10000 条数据，得到各类攻击样本检测情况，根据式(20)计算出各网络深度下的 G_{mean}，根据 G_{mean} 最大原则选用 DBN^4 (如图 3 中虚线所示)，具体检测结果如图 3。KELM 的参数 (γ,C) 设定为 $(2^1,2e^6)$，核函数为高斯核函数，如式（21）所示。

$$K(x,x_i)=\exp(-\gamma\|x-x_i\|^2),\ \gamma>0 \tag{21}$$

表 1　DBN 参数

Table 1　DBN parameter

网络深度	节点参数
DBN^1	122-30
DBN^2	122-100-30
DBN^3	122-100-70-30
DBN^4	122-110-100-60-30
DBN^5	122-110-100-70-50-30
DBN^6	122-110-100-80-60-40-30

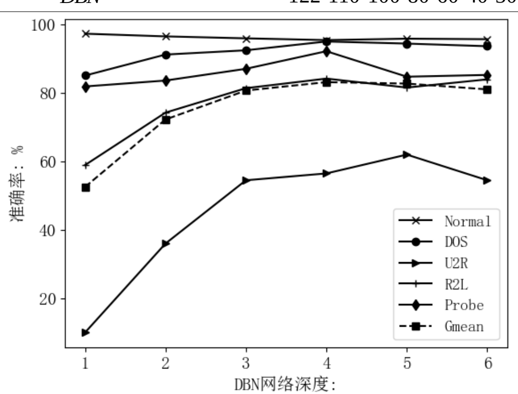


图 3　不同 DBN 深度下的检测结果

Fig. 3　Test results at different DBN depths

2.4　实验结果

2.4.1 DBN-WKELM 实验

为验证不同训练集分布情况下的检测情况，本次实验从训练集中抽取了 4 组不同样本分布的训练子数据集，从测试集中抽取 5 000 条，分别讨论了 DBN，DBN-KELM，以及改进后 DBN-WKELM 三种算法对 5 种攻击的检测情况以及检测结果的 G_{mean} 。实验数据分布情况如表 2 所示。

表 2　训练数据分布

Table 2　Training data distribution

	Normal	DOS	U2R	R2L	Probe
数据集 1	2650	2150	10	140	400
数据集 2	3000	2500	11	200	350
数据集 3	2000	3000	8	180	500
数据集 4	1200	1200	11	200	1500

各攻击样本的检测情况如表 3~6 所示，据式(20)计算出样本检测结果的 G_{mean} ，结果如图 4 所示。实验结果表明 DBN-KELM 算法比原始的 DBN 算法检测率高，对训练样本加权过后的算法 DBN-WKELM 算法在尽可能少的牺牲的大样本攻击类型的检测率，提高了对小样本类型攻击的检测率，在 4 个训练子数据集下检测结果的几何平均数 G_{mean} 均优于另外两种算法，弥补了传统入侵检测算法对大样本攻击类型识别率高而对小样本攻击识别率低的缺陷。

表 3　数据集 1 检测结果　/%

Table 3　Test result of data set 1　/%

算法	Normal	DOS	U2R	R2L	Probe
DBN	95.93	95.90	55.00	78.72	88.52
DBN-KELM	95.58	95.32	52.50	81.63	92.85
DBN-WKELM	95.54	95.43	64.00	88.31	93.76

表 4　数据集 2 检测结果　%

Table 4　Test result of data set 2　/%

算法	Normal	DOS	U2R	R2L	Probe
DBN	95.87	96.19	67.50	82.10	87.24
DBN-KELM	95.90	96.38	67.50	81.48	94.44
DBN-WKELM	95.44	95.82	83.50	90.52	90.62

表 5　数据集 3 检测结果　/%

Table 5　Test result of data set 3　/%

算法	Normal	DOS	U2R	R2L	Probe
DBN	95.25	97.01	62.50	81.59	88.43
DBN-KELM	95.18	97.73	69.00	82.61	94.30
DBN-WKELM	93.95	95.88	74.50	93.28	95.00

表 6　数据集 4 检测结果　/%

Table 6　Test result of data set 4　/%

算法	Normal	DOS	U2R	R2L	Probe
DBN	94.67	94.97	64.00	87.15	94.18
DBN-KELM	95.15	95.16	71.50	86.56	96.32
DBN-WKELM	93.94	94.85	85.00	90.49	94.88

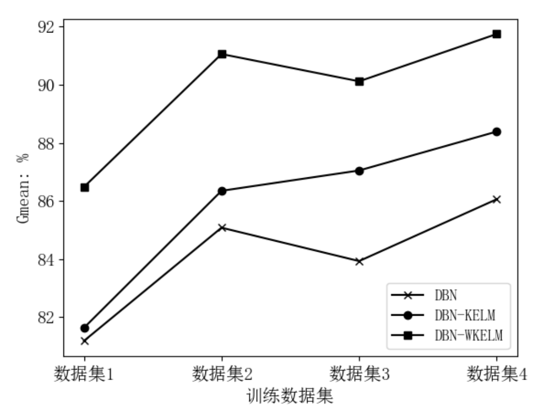


图 4　不同训练数据下的 G_{mean} 对比

Fig. 4　Comparison of G_{mean} under different training data

2.4.2 DBN-WOS-KELM 实验

本次实验中从 KDD99 的训练集抽取了 8000 条数据并分成 8 个数据块，从测试集中抽取了 5000 条数据（不含未知攻击形式），训练过程中将各个数据块经过 DBN 降维后送入序列核极限学习机(WOS-KELM)中训练，依次更新核极限学习机的输出权重 $\beta^{(i)}$，模拟现实入侵检测系统在线更新过程，各数据块的数据含量相同，具体分布情况如表 7 所示。训练集中 U2R 含量较少，采取自我复制的方法。

表 7　批量数据分布情况　/%

Table 7　The distribution of batch data　/%

攻击样本	Normal	DOS	U2R	R2L	Probe
数据含量	568	330	3	25	75

图 5 给出了随着训练批次增加各个攻击类型的检测变化情况，以及检测结果的 G_{mean} 的变化情况。实验结果表明随着训练批次的增加，各类攻击样本的检测率逐渐趋于稳定，检测结果的 G_{mean} 也依次增加，但增加的速率逐渐降低，分类器逐渐稳定。证明了本文所提出的 DBN-WOS-KELM 算法的有效性。

图 6 给出了 DBN-WOS-KELM 算法与 DBN-WKELM 算法在相同数据量下的训练时间曲线。实验结果表明在同等数据量的情况下 DBN-WOS-KELM 算法的训练效率明显优于 DBN-WKELM 算法。

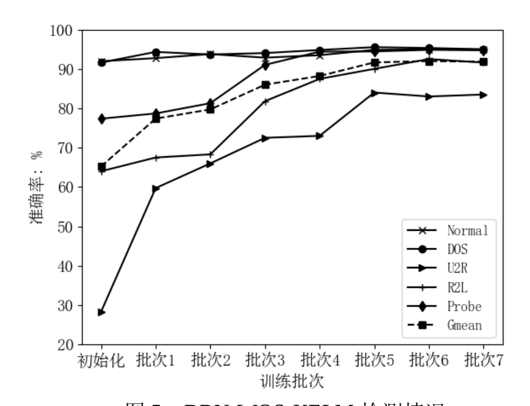


图 5　DBN-WOS-KELM 检测情况

Fig. 5　Detection situation of DBN-WOS-KELM

第一作者	导师/通信作者
姓名：汪洋	姓名:伍忠东
手机：15775137823	手机：13893455533
邮箱:1176601898@qq.com	邮箱：WUZHD@mail.lzjtu.cn
详细通信地址	邮编
甘肃省兰州市安宁区安宁西路 88 号 兰州交通大学	730070

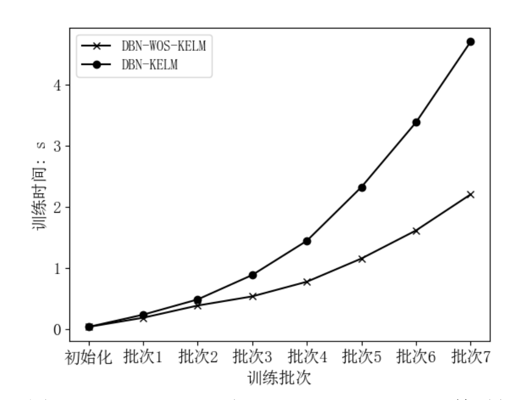


图 6　DBN-WKELM 与 DBN-WOS-KELM 训练时间

Fig. 6　Training time of DBN-WKELM and DBN-WOS-KELM

3　结束语

本文针对海量多源异构且数据分布不平衡的网络入侵检测问题以及传统深度学习算法无法根据实时入侵数据在线更新其输出权重的问题，提出了一种深度序列加权核极限学习的入侵检测算法(DBN-WOS-KELM 算法)。该算法结合了

DBN 特征提取和 KELM 快速学习的优势，不但通过样本加权解决了数据分布不均衡下的训练问题，在尽可能低地降低大样本攻击类别的识别率的前提下提高了对小样本攻击的识别率，而且能够在新的训练数据到来时在原输出权重的基础上在线更新其输出权重。最后在 KDD99 部分数据集上进行了实验，实验结果表明 DBN-WOS-KELM 算法不仅对各类型的攻击达到了较高的识别率，还能根据实际情况在线更新其输出矩阵，在同等数据量大小的情况下，训练效率优于 DBN-WKELM 算法。但本文仅仅使用了 KDD99 部分数据集验证了算法的可行性，下一步将考虑使用本文算法解决现实生活中的入侵检测问题。

参考文献：

[1] Gao Ni, Gao Ling, He yiyue, *et al*. Intrusion detection model based on deep belief network [J]. Journal of Southeast University:English Edition, 2015, 31(3): 339-346.)

[2] Ambusaidi M A, He Xiangjian, Nanda P, *et al*. Building an intrusion detection system using a filter-based feature selection algorithm [J]. IEEE Trans on Computers, 2016, 65(10): 2986-2998.

[3] 杨昆朋，基于深度学习的入侵检测 [D]. 北京:北京交通大学，2015. (Yang Kunpeng. Intrusion detection based on deep learning [D]. Beijing: Beijing Jiaotong University, 2015.)

[4] 逯玉婧. 基于深度信念网络的入侵检测算法研究 [D].石家庄：河北师范大学, 2016. (Lu Yujing. Research on intrusion detection algorithm based on deep belief network [D]. Shijiazhuang:Hebei Normal University, 2016.)

[5] Zong Weiwei, Huang Guangbin, Chen Yiqiang. Weighted extreme learning machine for imbalance learning [J]. Neurocomputing, 2013, 101: 229-242.

[6] Mirza B, Lin Zhiping, Toh K A. Weighted online sequential extreme learning machine for class imbalance learning [J]. Neural processing letters, 2013, 38(3): 465-486.

[7] Ding Shuya, Mirza B, Lin Zhiping, *et al*. Kernel based online learning for imbalance multiclass classification [J]. Neurocomputing, 2018, 277: 139-148.

[8] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets [J]. Neural computation, 2006, 18(7): 1527-1554.

[9] Hinton G E. Training products of experts by minimizing contrastive divergence [J]. Neural computation, 2002, 14(8): 1771-1800.

[10] Hinton G E. A practical guide to training restricted Boltzmann machines [M]//Neural networks: Tricks of the trade. Berlin:Springer,2012: 599-619.

[11] Deng WanYu, Ong Y S, Tan P S, *et al*. Online sequential reduced kernel extreme learning machine [J]. Neurocomputing, 2016, 174: 72-84.

[12] Dhanabal L,Shantharajah S P. A study on NSL-KDD dataset for intrusion detection system based on classification algorithms [J]. International Journal of Advanced Research in Computer and Communication Engineering, 2015, 4(6): 446-452.

[13] Al Helal M, Haydar M S, Al Mostafa S M.. Algorithms efficiency measurement on imbalanced data using geometric mean and cross validation [C]//Proc of International Workshop on Computational Intelligence. Piscataway, NJ: IEEE Press, 2016:110-114.